AAS 95-048

AMERICAN ASTRONAUTICAL SOCIETY

Scene Simulation for Real-Time Testing of the Cassini Star Tracking and Identification Functions

James W. Alexander and Daniel H. Chang
Jet Propulsion Laboratory, California Institute of Technology

18th ANNUAL AAS GUIDANCE AND CONTROL CONFERENCE

February 1-5, 1995
Keystone, Colorado

Sponsored by
ROCKY MOUNTAIN SECTION
AMERICAN ASTRONAUTiCAL SOCIETY

AAS **Publications Office,** P.O. **Box 28130- San Diego, California 92198**

CASSINI MISSION SUM MARY

<u>The challenge of the Cassini mission</u> - to return to the Saturnian system and perform detailed observations of the planet, its moons and rings – observations that earlier Voyager fly-by missions were not able to accomplish.

The Cassini spacecraft consists of an orbiter, which carries a dozen scientific instruments, and the ESA-provided Huygens Probe. The probe will be released by the orbiter into the atmosphere of Titan, and will use the orbiter to relay data to the Earth.

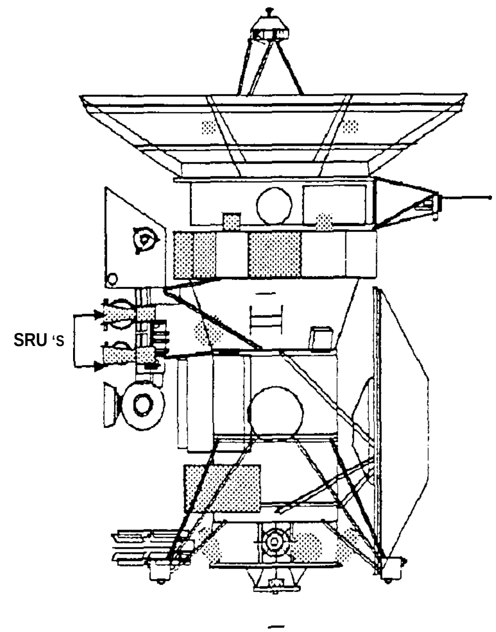
Cassini is scheduled to be launched in October, 1997 on a Titan IV/Centaur launch vehicle, and arrive in June of 2004, and will spend nearly four years making scientific observations.

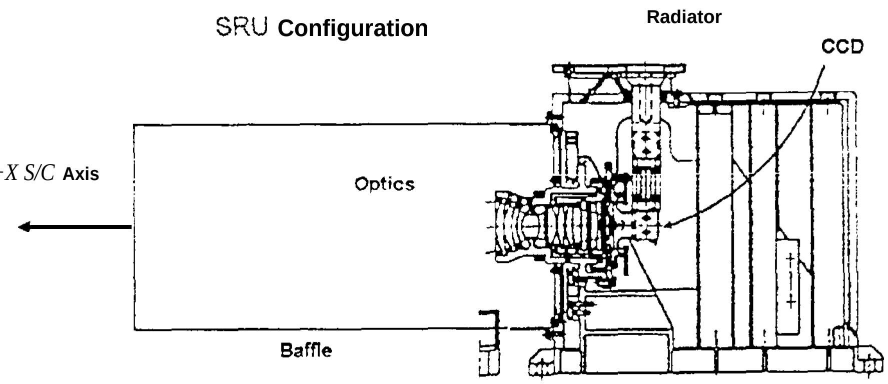
The STELLAR REFERENCE UNIT (SRU)

The Cassini Attitude and Articulation Control System uses a single computer to perform attitude measurements and control functions, including the processing of the raw pixel data from the SRU. This integrated approach allows combining several functions which have traditionally been separate, such as allowing the predicted attitude to be used by the SRU star tracking algorithm software to re-acquire known stars after a long turn through the Sun or Saturn. The Cassini spacecraft and the location of the two co-foresighted (for redundancy) SRUS are shown at the right.

<u>SRU kev features:</u>

- The SRU uses a 1024^2 Loral CCD, with both MPP and Partially Inverted modes.
- Pixels are transmitted through a 1 Mbit bus, the *pixel bus*, to the flight computer. A test port on the SRU allows pixels from another source to be substituted. (The rest of the poster deals with this).
- The FOV is 15 degrees square.
- Exposure times are commendable in 1 ms increments.
- Has up to 5 track windows, each of which can be placed at an arbitary position and size. Typical size is 20x20 pixels (5 mrad 2), with a centroid area of 5x5.
- On-chip CCD charge summation is commendable from 1xl to 4x16 to allow for faster readout times during acquisition and when the S/C is turning. Additionally, the number of pixels processed in the flight computer is reduced.





SRU Configuration

The Star Tracking Functions

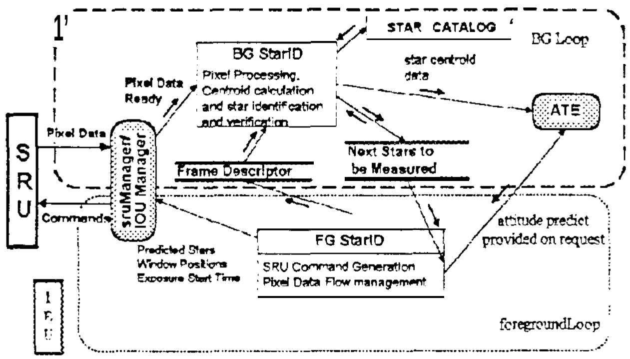
The software that controls the S R U is referred to as StarID. The prime functions of StarID are attitude initialization, track, end (re-)acquisition. StarID is fully autonomous, and as such, the entire star catalog Is on-board (approximately 4000 stars, 15 to 20 per FOV), and Includes magnitude, color, and hints as to the separation of the star from possible confusing stars. With the 15 degree FOV, the star catalog goes as deep as about S.2 magnitude to give full sky coverage.

The attitude initialization function determines the S/C attitude after launch, or under fault conditions. In this mode, Starl D commands the SRU to deliver images that cover the entire FOV, scans the resulting pixels for bright spots, then matches to the on-board catalog, requiring a minimum of 4 or 5 matching stars. While Cassini requirements allow the use of the sun position, the algorithm could be used for all sky identification (but is not optimized for such).

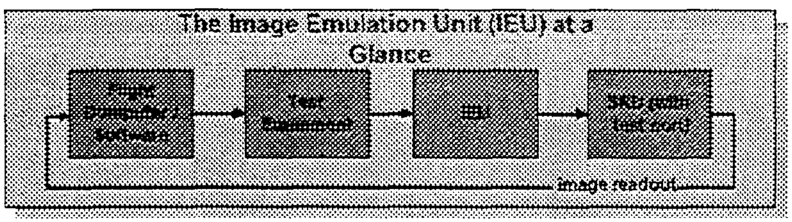
Track depends on predicts from the attitude estimator. Using this estimated attitude, the track function commands up to 5 track windows, placing each about calculated star positions. The typical track mode window k Implemented as 5 mrad to 40 mred square reedout area, depending on the rate and uncertainty. Additionally, track verities the stars by comparing magnitudes and star pairwise separat Ions.

Finally, the acquisition mode is used for recovery when stars ara lost, such as after a quick turn, but the uncertainty Is less than a few degrees.

Because of all these modes of operation, testing is a large issue. While field tests are imports nt, scenarios that *can* be tested are limited. For Cassini, it became important to devise a way to test the algorithms in both the unit tests, and when the complete fright software is integrated with the flight hardware. The approach that JPL has taken for Cassini is described on this poster. It provides the method for testing flight code without patches within the code, or artifical constraints which limited testing In the past.



The light connection between StarID and the attitude esltmator, which uses gyro data, sun sensor data, etc. 10 enhance the attitude estimate, is shown in the above diagram.



- **Purpose:** provide a high fidelity simulation of the Cassini SRU
  - . Functional fidelity: designed for speed. Can be used for real-time, hardware-in-the-loop simulations to support subsystem integration
  - . Performance fidelity: used In the design and debugging of StarID and tracking algorithms

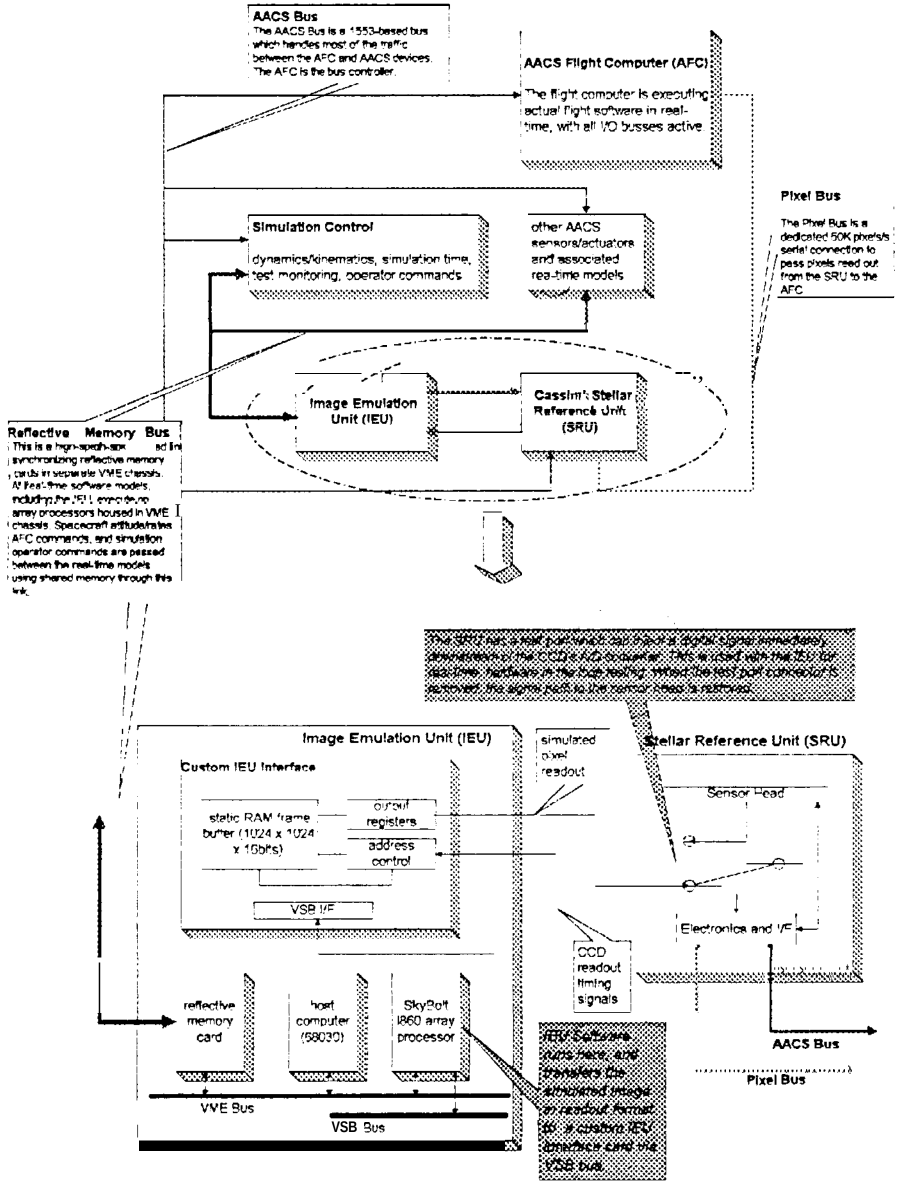
- **Features**
  - Complete simulation of all SRU modes
    - simulates the windowed readout and CCD analog summation modes, A/D gain, MPP mode, A/D offset, and anti-blooming when commanded.
    - . mimics firmware state machine, AACS Bus command Interface (RTIOU)
  - Detalled optical model for point sources (stars)
    - Includes effects of geometric distortion, realistic point spread function, and streaking due to spat-raft motion during commanded exposures
  - Relevant CCD and noise effects modeled
    - . CCD read-noise, dark current, and photon arrival statistics light
    - . full-well blooming, readout streaking (due to SRU lacking a shutter)
    - . proton hit flashes, damaged pixels, dark current "hot pixels"
  - . Basic extended body (planets/moons, rings, sun) simulation
  - . Star catalog
    - . deeper than flight catalog ~25000 stars down to -7Mv compared to 4000 for flight
    - can change contents In real time to model variable stars, flight-catalog error% etc.

- **Model /Simulation verified by comparison against *real* sky**
  - . Data from field tests used to verify Simulation

How the Image Emulation Unit (IEU) Is Used

The IEU fulfills two basic needs in the development of the Altitude and Articulation Control System (AACS) on the Cassini spacecraft. During flight software development, a high-fidelity model of the SRU is needed to incrementally test the star identification and tracking algorithm design. During subsystem and spacecraft integration, the AACS flight software and hardware are exercised in real-time, closed-loop fashion, also requiring a high fidelity SRU model, but running in real time. Software has been developed which supports both uses with minimal modification.

Subsystem and Spacecraft Integration Hardware-in-the-Loop Testing



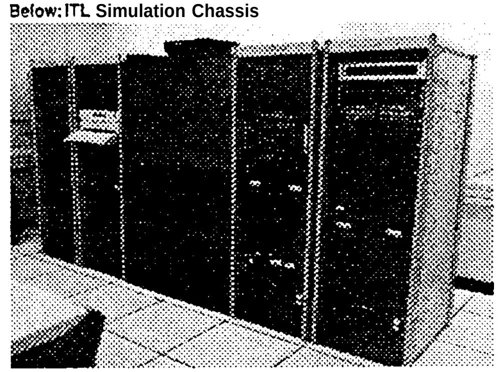
Key Aspects of Testing

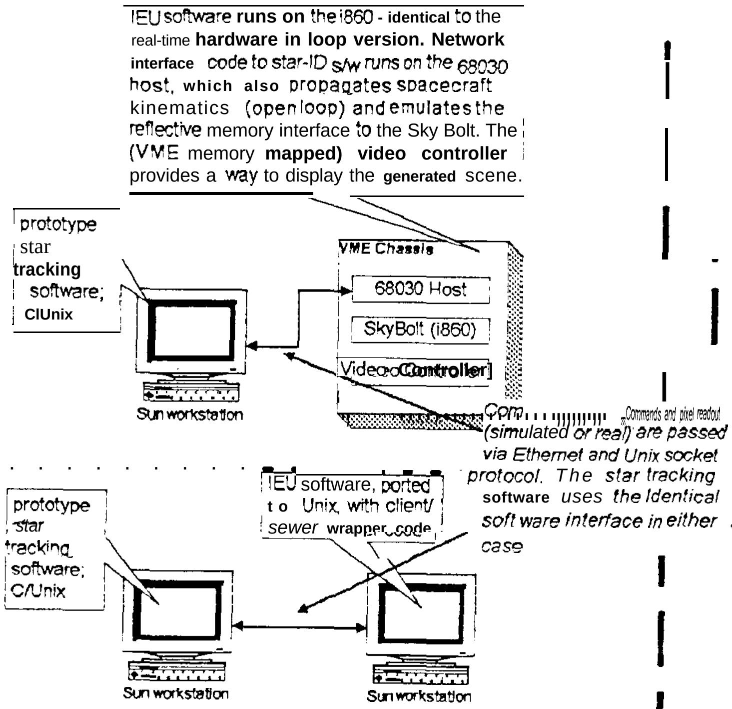
- **Hardware Design/Analysis**
  - Characterization and testing of SRU hardware
  - Performed by Officine Galileo and JPL
- **Algorithm Development**
  - Standalone testing of star ID and tracking flight software design
  - Validate functionality, performance
  - Workstation environment, non real-time
- **Real-Sky Testing**
  - Performed at Table Mountain Observatory
  - Uses SRU breadboard and engineering units
  - Uses Prototype flight software
- **Subsystem Integration**
  - Cassini AACS Integration Test Lab environment
  - Real-time, hardware in the loop simulations

IEU directly supports Algorithm Development and Subsystem Integration testing. Effort in transition to Real-Sky testing is minimized by using identical software interfaces.

Above: Cassini AACS Integration Test Lab (ITL), hardware integration arm

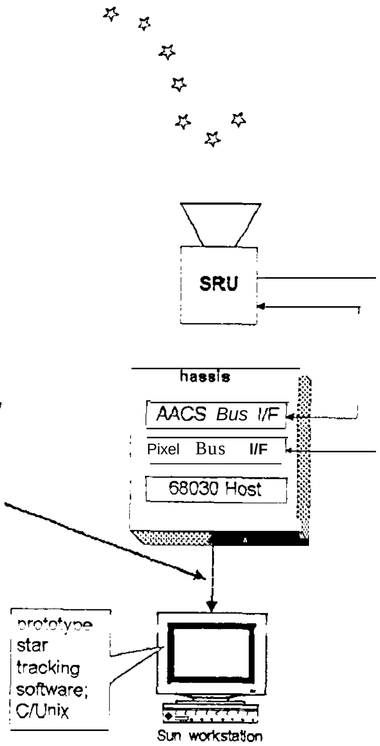
Below: ITL Simulation Chassis





IEU Simulation Supports Star Tracking Software Development

The IEU software developed for hardware in the loop testing in the Cassini Integration Test Lab is ported with minimal changes to support development of the star tracking flight software. In place of the AACS Bus and the Pixel Bus, conventional Ethernet and Berkeley socket protocol is used. This allows for great flexibility as well as approximate real-time testing. The Unix version of the IEU is used for maximum convenience, while the i860 version is *used for* more rigorous real-time simulation.



Real Sky Testing at Table Mountain Observatory

The prototype star tracking software is thoroughly tested under the real sky using the Engineering Model of the SRU to ensure proper operation. The software interfaces to the IEU and the SRU are identical.

Table Mountain Observatory